Department of Computer Science

University of Engineering and Technology, Lahore

Write the C++ Functions of the following problems.

Problem 1: Aeroplane Seat Reservation System

Write a program that can be used to assign seats for a commercial aeroplane. The aeroplane has 13 rows, with six seats in each row. Rows 1 and 2 are first class, rows 3 through 7 are business class, and rows 8 through 13 are economy class. Your program must prompt the user to enter the following information:

1. Ticket type (first class, business class, or economy class)
2. Desired seat ((1-13) for Rows and (A to F) for Columns)

Output the seating plan in the following form:

	A	B	C	D	E	F
Row 1	*	*	X	*	X	X
Row 2	*	X	*	X	*	X
Row 3	*	*	X	X	*	X
Row 4	X	*	X	*	X	X
Row 5	*	X	*	X	*	*
Row 6	*	X	*	*	*	X
Row 7	X	*	*	*	X	X
Row 8	*	X	*	X	X	*
Row 9	X	*	X	X	*	X
Row 10	*	X	*	X	X	X
Row 11	*	*	X	*	X	*
Row 12	*	*	X	X	*	X
Row 13	*	*	*	*	X	*

Here, ***** indicates that the seat is available; **X** indicates that the seat is occupied. Make this a menu-driven program; show the user's choices and allow the user to make the appropriate choices. When the program is about to end, store the currently reserved seats arrangement in the file. Next time the program starts, read the file and start from the previously stored reserved seats.

Make the following functions for the menu of the system

1. Load the reserved seats arrangement from the file
2. Ask the user for the ticket type and desired seat

3. Print the reserved seats arrangement
4. Store the reserved seats arrangement in the file
5. New Plane seats arrangement.

Problem 2: Matrix Calculator (Important for Video Profile Activity)

Ma Sha Allah!!! You have become an excellent programmer. Now, a mathematician has hired you to develop a software that can apply different operations on Matrices.

Write different functions to perform various Matrix operations on a given 2-D Arrays.

1. **Addition of two matrices (A+B)**

   For Adding two matrices, order of the matrices must be the same

$$\begin{bmatrix} 4 & 8 \\ 3 & 7 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 5 & 2 \end{bmatrix} = \begin{bmatrix} 4+1 & 8+0 \\ 3+5 & 7+2 \end{bmatrix}$$

2. **Subtraction of two matrices (A-B)**

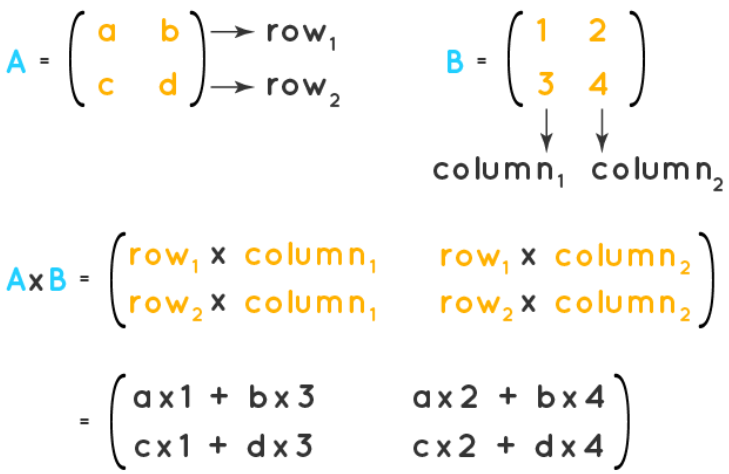
   For Subtracting two matrices, order of the matrices must be the same

$$A = \begin{bmatrix} 12 & -3 \\ 2 & 15 \end{bmatrix} \text{ and } B = \begin{bmatrix} 6 & 1 \\ 11 & -8 \end{bmatrix}$$

$$A - B = \begin{bmatrix} 12-6 & -3-1 \\ 2-11 & 15-(-8) \end{bmatrix} = \begin{bmatrix} 6 & -4 \\ -9 & 23 \end{bmatrix}$$

3. **Multiplication of two matrices (A*B)**

   For multiplying two matrices, The number of columns in the first matrix must be equal to the number of rows in the second matrix.



4. **Scalar multiplication (value*Matrix)**

   For scalar multiplication, The matrix can be of any order. You just have to multiply all elements in the matrix by the scalar.

$$2 \cdot \begin{bmatrix} 10 & 6 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 2 \cdot 10 & 2 \cdot 6 \\ 2 \cdot 4 & 2 \cdot 3 \end{bmatrix}$$

5. **Is Identity matrix (isIdentity(Matrix))**

   In linear algebra, the identity matrix of size n is the n × n square matrix with ones on the main diagonal and zeros elsewhere.

$$I_1 = [1],\ I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},\ I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},\ \ldots,\ I_n = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

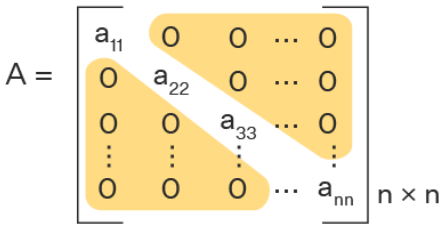
6. **Transpose of the Matrix (transpose(Matrix))**

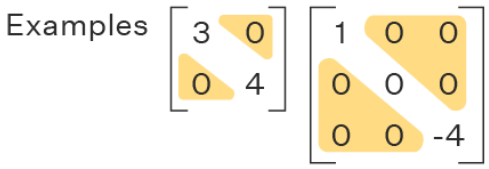
   In linear algebra, the transpose of a matrix is actually an operator that flips a matrix over its diagonal by switching the row and column indices of matrix B and producing another matrix.

$$A = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}_{2 \times 3} \qquad A^T = \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}_{3 \times 2}$$

7. **Is Diagonal Matrix (isDiagonal(Matrix))**

   A diagonal matrix is a square matrix where all of its non-diagonal elements are zero.





8. **Is Symmetric Matrix (isSymmetric(Matrix))**

   In linear algebra, a symmetric matrix is a square matrix that is equal to its transpose.

$$B = \begin{bmatrix} 2 & 3 & 6 \\ 3 & 4 & 5 \\ 6 & 5 & 9 \end{bmatrix}$$

$$B^T = \begin{bmatrix} 2 & 3 & 6 \\ 3 & 4 & 5 \\ 6 & 5 & 9 \end{bmatrix}$$

There are 3 arrays available with names A, B and C each of 3x3 size. You can initialize any array any time.

The program runs until the user enters “Exit”.

The format to take input is as follows:

A={1,2,3,4,5,6,7,8,9}

C={1,0,0,0,1,0,0,0,1}

This will initialize a **3x3** array.

You will take input in a string format and then extract the information of the array elements and store them in the array accordingly.

The user will enter **C=A+B** for matrix addition, **C=A-B** for matrix subtraction, **C=A*B** for matrix multiplication and so on and store the result in C array. Results will always be stored in the **C** array.

You have to take input in a string and then separate out what is the first matrix, what is the operation and what is the second matrix.

print(array) will print the contents of the array on the console.

E.g, **print(A)**, **print(B)**, **print(C)**

For checking different properties of the matrices, the user will also write the function call on the console i.e., **isIdentity(A)**, **isDiagonal(B)**, **isSquare(A)**.

You have to take that input in the string format and separate out the information accordingly.

Sample Output:

```
>>A={1,0,0,0,1,0,0,0,1}
>>print(A)

1       0       0
0       1       0
0       0       1

>>B={1,1,1,1,1,1,1,1,1}
>>print(B)

1       1       1
1       1       1
1       1       1

>>C=A+B
>>print(C)

2       1       1
1       2       1
1       1       2
```

```
>>C=A*B
>>print(C)

17      17      17
17      17      17
17      17      17

>>isDiagonal(B)

0
>>isDiagonal(A)

1
>>isIdentity(A)

1
>>transpose(A)
>>print(C)

1       0       0
0       1       0
0       0       1

>>Exit
PS C:\C++\Week11>
```